

## DESAIN DAN IMPLEMENTASI PURWARUPA PENDETEKSI DINI KEBAKARAN GEDUNG MENGGUNAKAN APLIKASI MOBILE BERBASIS ANDROID DAN INTERNET OF THINGS (IOT)

### DESIGN AND IMPLEMENTATION OF FIRE BUILDING DETECTION PROTOTYPE USING ANDROID BASED MOBILE APPLICATION AND INTERNET OF THINGS (IOT)

Alya Sekar Ayu<sup>1</sup>, Ir. Ahmad Tri Hanuranto, M.T.<sup>2</sup>, Atik Novianti, S.ST., M.T.<sup>3</sup>

<sup>1,2,3</sup>Prodi S1 Teknik Telekomunikasi, Fakultas Teknik, Universitas Telkom

<sup>1</sup>alyasekarayu@student.telkomuniversity.ac.id, <sup>2</sup>athanuranto@telkomuniversity.ac.id,

<sup>3</sup>atiknovianti@tass.telkomuniversity.ac.id

#### Abstrak

Deteksi dini kebakaran yang bisa membuat tindakan dan keputusan yang cepat sangat penting dalam sistem pemadam kebakaran. Namun pada kenyataannya sistem pendeteksi yang sudah banyak digunakan oleh masyarakat masih sebatas *alarm* yang hanya bisa didengar ketika pemilik rumah sedang berada di tempat. Oleh karena itu, *Internet of Things* menjadi salah satu alternatif yang mampu menyediakan sistem cerdas untuk kebutuhan tersebut.

Sistem yang dibuat pada penelitian ini bertujuan mencegah atau meminimalisir kerugian dari kebakaran sehingga dapat diketahui dan ditanggulangi lebih awal. Pada purwarupa yang dibuat untuk pendeteksi kebakaran gedung ini, terdiri dari dua sub bagian yaitu alat dan aplikasi *mobile* yang saling terintegrasi. Pada penelitian ini hanya terfokus pada perancangan aplikasi *mobile* Android. Sebuah gedung dari jarak jauh akan dimonitor melalui aplikasi *Android* oleh *user* atau pemilik gedung yang terhubung melalui internet pada alat. Platform *ThingSpeak* digunakan sebagai *database* untuk menyimpan dan mengirim data sensor dari alat menuju aplikasi *Android* yang kemudian akan mengirimkan notifikasi kepada *user* atau pemilik gedung.

Dari hasil pengujian sistem, diketahui bahwa aplikasi *mobile* dapat terhubung dengan *database ThingSpeak* dan pembacaan data *database* berjalan dengan baik. Didapatkan nilai rata-rata *delay end to end database* menuju aplikasi *mobile* sebesar 7,7s. Sesuai dengan standar ITU-T G.1010 untuk *delay (delay end to end < 10 s)* nilai tersebut dikategorikan baik. Pada pengujian aplikasi juga diketahui ketika kondisi waspada akan mengirimkan notifikasi kepada *user* atau pemilik gedung berupa peringatan dini, dan ketika kondisi bahaya juga akan mengirimkan notifikasi kepada *user* atau pemilik gedung berupa peringatan dini dan otomatis melakukan panggilan kepada petugas pemadam kebakaran.

**Kata kunci :** *IoT, Mobile Android, ThingSpeak, Delay.*

#### Abstract

Early detection of fires that can make quick action and decisions is essential in a fire fighting system. However, in reality the detection system that is commonly used by the community is still limited to an alarm that can only be heard when the owner of the house is present. Therefore, the Internet of Things is an alternative that is able to provide smart systems for these needs.

The system created in this study aims to prevent or minimize losses from fires so that they can be detected and addressed early. Thus system consists of two sub-parts, such as tool and mobile applications which are integrated with each other. In this study only focused on designing an Android mobile application. A building will be remotely monitored via the Android application by the user or building owner connected through the internet from the device. The ThingSpeak platform is used as a database to store and send sensor data from the device to the Android application which then sends a notification to the user or building owner.

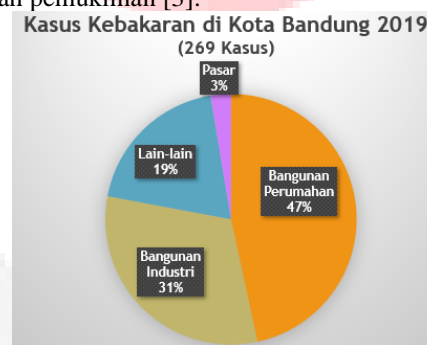
From the results of system testing, it is known that the mobile application can connect with the ThingSpeak database and the reading of the database data runs well. Obtained the average value of end to end database delay to the mobile application is 7,7s. In accordance with the ITU-T G.1010 standard for delay (delay end to end <10 s) the value is categorized as good. In application testing, it

is also known when the condition is alert, it will send a notification to the user or building owner in the form of an early warning, and when the condition is in danger, it will also send a notification to the user or building owner in the form of an early warning and automatically make a call to the fire fighter.

**Keywords:** *IoT, Mobile Android, ThingSpeak, Delay.*

## 1. Pendahuluan

Deteksi dini kebakaran dapat menyelamatkan manusia serta mengurangi kerugian sebanyak mungkin. Kebakaran sering terjadi di berbagai tempat seperti rumah, pabrik, kompleks komersial, rumah sakit dan tempat umum lainnya sehingga diperlukannya sistem pemadam kebakaran yang cerdas berbasis Internet of Things (IoT) [1]. Ini menandakan bahwa generasi terbaru internet dibutuhkan untuk semua hal dalam kehidupan manusia sehingga semua hal dalam kehidupan kita dapat terhubung ke internet. Dengan demikian manusia akan memiliki kemampuan untuk berkomunikasi dan berinteraksi satu sama lain untuk mengirim dan menerima data sehingga bisa melakukan fungsi-fungsi spesifik melalui jaringan [2]. Berdasarkan data (Dinas Pencegahan dan Penanggulangan Kebakaran Kota Bandung) tahun 2019, terjadi 269 kebakaran dengan sekitar 120 (47%) kejadian terjadi di daerah pemukiman [3].



**Gambar 1.1** Kejadian Kebakaran di Kota Bandung Tahun 2019

Eltom, dkk [5] membuat sistem pendeteksi kebakaran dini berbasis IOT dengan beberapa sensor *input* seperti DHT-11, MQ2, dan sensor api yang diproses melalui Arduino dan dipantau melalui *web server* dan aplikasi Cayenne. Ketika sistem mendeteksi adanya kebakaran, maka aplikasi *mobile* akan mengirimkan SMS kepada nomor *user* atau pengguna aplikasi *mobile* tersebut.

Berdasarkan dengan uraian masalah pada Gambar 1.1 maka penting untuk menekan angka terjadinya kebakaran dengan membuat sistem keamanan kebakaran yang efisien berbasis IoT menggunakan Arduino mega 2560 yang dipantau melalui aplikasi mobile. IoT adalah teknologi baru yang cocok untuk aplikasi pemadam kebakaran karena IoT memiliki tingkat kecerdasan, skalabilitas tinggi, dan kemampuan berbagi sumber daya yang tinggi. [4]. Mikrokontroler bisa digunakan sebagai pendeteksi dini sistem pemadam kebakaran berbasis IoT karena dapat dipantau melalui jarak jauh dengan bantuan komputer sebagai server dan aplikasi mobile untuk komunikasi kepada pengguna [6]. Data sensor dari mikrokontroler disimpan didalam basis data yang kemudian diambil oleh aplikasi mobile dari komputer server. Maka dari itu sistem Smart Fire Alarm (Fi-Real) berbasis IoT dapat diimplementasikan sebagai solusi dalam menangani masalah peristiwa kebakaran yang lebih efisien.

Pada sistem Fi-Real memiliki kemampuan untuk menangani peristiwa kebakaran segera setelah aplikasi mobile mendapatkan data sensor dari alat. Tujuannya untuk mengurangi kerusakan dan kerugian yang disebabkan oleh kebakaran yang mungkin bukan hanya kerugian material, tetapi bisa lebih buruk lagi yaitu kerugian korban jiwa. Sistem ini dibuat untuk memantau parameter yang berpengaruh terhadap kemungkinan terjadinya kebakaran pada suatu gedung dengan mengimplementasikan konsep IoT.

Sistem Fi-Real terbagi menjadi dua sub bagian yaitu alat dan aplikasi mobile yang saling terintegrasi. Pada tugas akhir ini hanya terfokus pada perancangan aplikasi mobile. Dengan sistem Fi-Real yang dirancang ini, pemilik gedung dapat memantau keadaan dari masing-masing ruangan dari suatu gedung, sehingga diharapkan bisa meminimalisir kasus terjadinya kebakaran di masa depan. Maka dari itu diharapkan sistem ini mampu menjawab dari persoalan peristiwa kebakaran yang terjadi selama ini.

## 2. Dasar Teori

### 2.1 Kebakaran

Kebakaran adalah munculnya nyala api yang diakibatkan oleh proses oksidasi eksotermis yang berlangsung cepat. Terciptanya kebakaran dikarenakan ada bahan bakar atau material yang dapat menyulut besarnya kobaran api. Terdapat beberapa faktor yang menjadi penyebab kebakaran gedung, antara lain [4] :

1. Kebocoran gas
2. Puntung rokok yang dibuang sembarangan
3. Hubungan arus pendek listrik
4. Bermain api di ruangan yang mudah terbakar

### 2.2 Android

Android adalah sistem operasi untuk perangkat seluler smartphone berbasis Linux. Android juga memiliki kelebihan dibandingkan dengan sistem operasi mobile lain yaitu open source dan telah didukung standar dan penerbitan API (Application Programming Interface) yang dimanfaatkan secara keseluruhan dengan biaya relative lebih murah. Kemudahan dalam membangun aplikasi Android memberikan akses untuk seluruh library [7].

### 2.3 Android SDK

Android SDK adalah tools API (Application Programming Interface) yang diperlukan untuk memulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, middleware, dan aplikasi kunci yang di release oleh google. [8]

### 2.3 Kotlin

Kotlin merupakan bahasa pemrograman statis yang berjalan pada Java Virtual Machine dan dapat dikompilasi ke kode JavaScript. Kotlin pertama kali dikembangkan oleh tim programmer JetBrains di Saint Petersburg, Russia. Kotlin pertama kali dikenalkan pada tahun 2010 untuk penggunaan pribadi, pada tahun 2011 dengan mendapatkan lisensi apache maka kotlin resmi menjadi open source. Pada tahun 2015 Google mencoba Kotlin pada Android Studio, pada tahun 2017 Google IO Conformance mengumumkan Kotlin sebagai bahasa pemrograman android secara resmi. Kotlin merupakan bahasa pemrograman yang selangkah lebih maju dibandingkan dengan Java untuk pengembangan Android. (Panchal & patel, 2017). Dapat dilihat perbandingan bahasa pemrograman java dengan kotlin pada table 2.2. [9]

### 2.4 ThingSpeak

ThingSpeak merupakan server Internet of Things yang cukup diminati karena konfigurasinya yang mudah. Untuk memulai ThingSpeak pertama-tama kita harus membuat akun pada server tersebut dengan alamat email yang kita miliki. Setelah mendapat akun pada server tersebut, maka Langkah selanjutnya ialah mengatur dashboard atau tampilan awalnya dengan perangkat yang akan digunakan untuk melakukan pengiriman data. [10]

### 2.5 Internet of Things (IoT)

Internet of Things (IoT) merupakan sebuah konsep yang bertujuan untuk memperluas manfaat dari konektivitas internet yang tersambung secara terus-menerus yang memungkinkan kita untuk menghubungkan mesin, peralatan, dan benda fisik lainnya dengan sensor jaringan dan actuator untuk memperoleh data dan mengelola kinerjanya sendiri, sehingga memungkinkan mesin untuk berkolaborasi dan bahkan bertindak berdasarkan informasi baru yang diperoleh secara independent. [11]

### 2.6 UI/UX

User Interface (UI) merupakan desain tampilan antarmuka sebuah sistem, baik itu aplikasi, software, web, komputer dan peralatan elektronik lainnya. UI mencakup tata letak dan desain grafis suatu aplikasi seperti tombol yang akan diklik oleh pengguna, teks, gambar, text entry fields, serta semua item yang berinteraksi dengan pengguna. Semua hal tersebut diatur sedemikian rupa agar dapat memaksimalkan pengalaman pengguna. Sedangkan user experience adalah proses meningkatkan kepuasan pengguna dalam interaksi antara pengguna dan produk agar mudah untuk digunakan. Sehingga UX berfokus pada bagaimana UI dapat bekerja dengan baik [12].

### 2.7 White-Box Testing

White Box Testing adalah pengujian detail pada struktur internal dan kode. Pada pengujian ini penguji harus memiliki pengetahuan penuh tentang internal dan struktur kode [13].

### 2.8 Black-Box Testing

Black Box Testing adalah teknik pengujian tanpa memiliki pengetahuan dari kerja internal aplikasi. Hanya memeriksa aspek-aspek dasar dari sistem dan tidak memiliki relevansi dari struktur internal sistem [13]. Pengujian Blackbox memfokuskan pada keperluan fungsionalitas dari perangkat lunak.

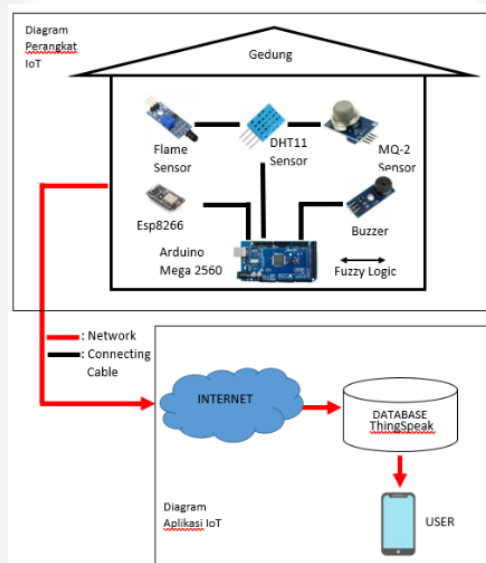
### 2.9 Delay

Delay adalah waktu yang dibutuhkan sebuah paket dari saat dikirim hingga ke tujuannya. Dalam penelitian ini delay dihitung dari mulai aplikasi mobile mengirimkan perintah dan input hingga data tersimpan ke Firebase, dan sebaliknya aplikasi mobile memanggil data dari Firebase ke aplikasi mobile [14]

## 3. Perancangan Sistem

### 3.1. Desain Sistem

pengerjaan Tugas Akhir ini penulis terfokus pada pengembangan aplikasi Mobile. Pada desain sistem ini terdapat dua diagram yaitu diagram perangkat IoT dan diagram aplikasi IoT. Pada diagram perangkat IoT berisikan sensor-sensor yang digunakan pada alat, dan pada diagram aplikasi IoT terdapat platform dan output dari hasil perancangan tersebut

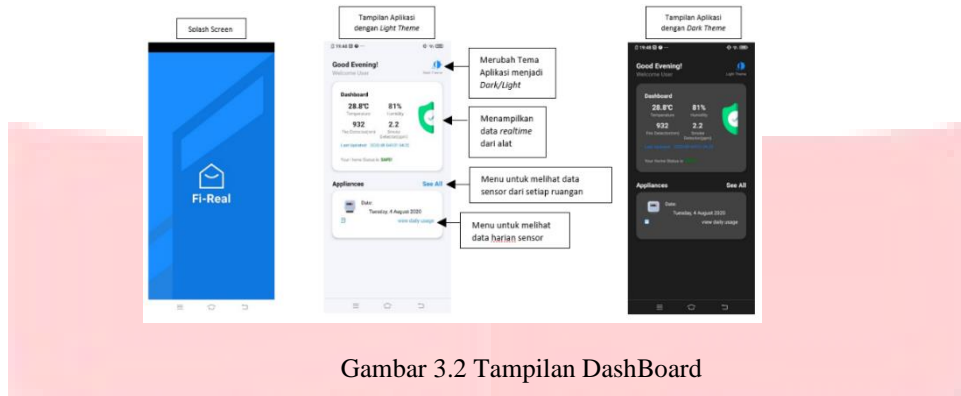


Gambar 3.1 Desain sistem.

Pada gambar 3.1 menjelaskan desain sistem secara keseluruhan. Desain sistem tersebut dirancang untuk mendeteksi kebakaran sejak dini pada sebuah gedung berbasis algoritma fuzzy logic yang akan menentukan sebuah keputusan yang akurat berdasarkan parameter-parameternya. Lalu output pada sistem ini yaitu mengirimkan sebuah informasi yang efisien, simple, dan relevan yang berupa notifikasi pada aplikasi mobile kepada pemilik gedung dan petugas pemadam kebakaran. Selain itu juga dapat dipantau melalui aplikasi mobile jika terdeteksi adanya kemungkinan kebakaran atau keadaan berbahaya.

### 3.2 Desain UI Aplikasi Mobile

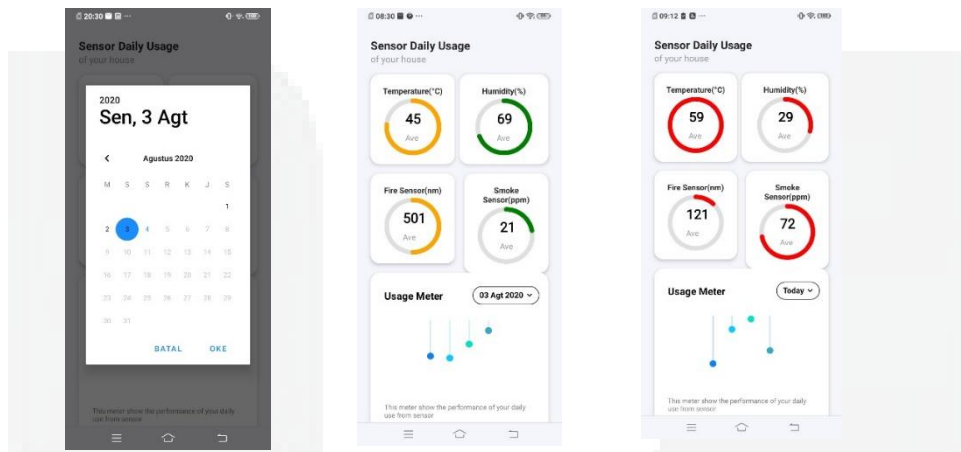
#### 3.2.1 Tampilan Dashboard



Gambar 3.2 Tampilan Dashboard

Pada tampilan ini menampilkan beberapa fitur seperti fitur untuk merubah tema aplikasi. Lalu pada dashboard menampilkan data realtime setiap sensor dari alat yang sudah terintegrasi dengan ThingSpeak. Selain menampilkan data sensor, dashboard juga menampilkan status gedung (aman, waspada, bahaya), menu untuk melihat data penggunaan harian sensor, dan data sensor dari setiap ruangan.

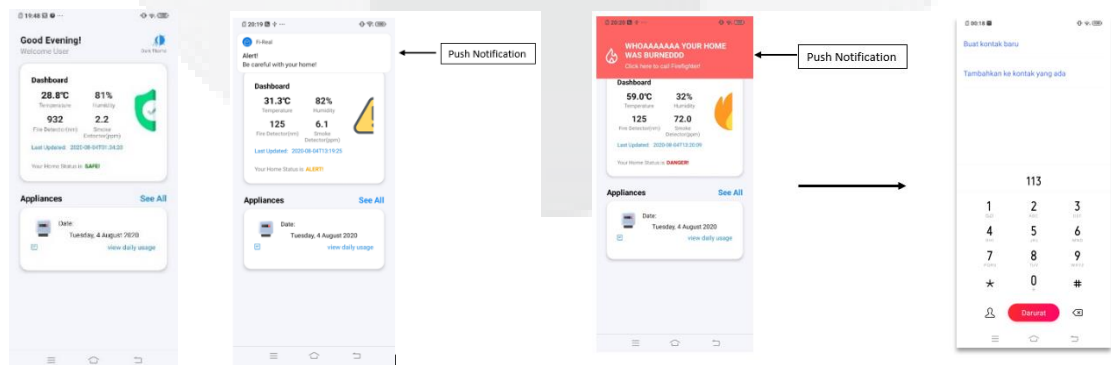
#### 3.2.2 Menu Sensor Daily Usage



Gambar 3.3 Menu Sensor Daily Usage

Pada menu ini bertujuan untuk menampilkan nilai rata-rata sensor setiap harinya. Ada tiga indikator warna yang menunjukkan keadaan sensor yaitu warna merah yang tandanya bahaya, oranye tandanya waspada, dan hijau tandanya aman.

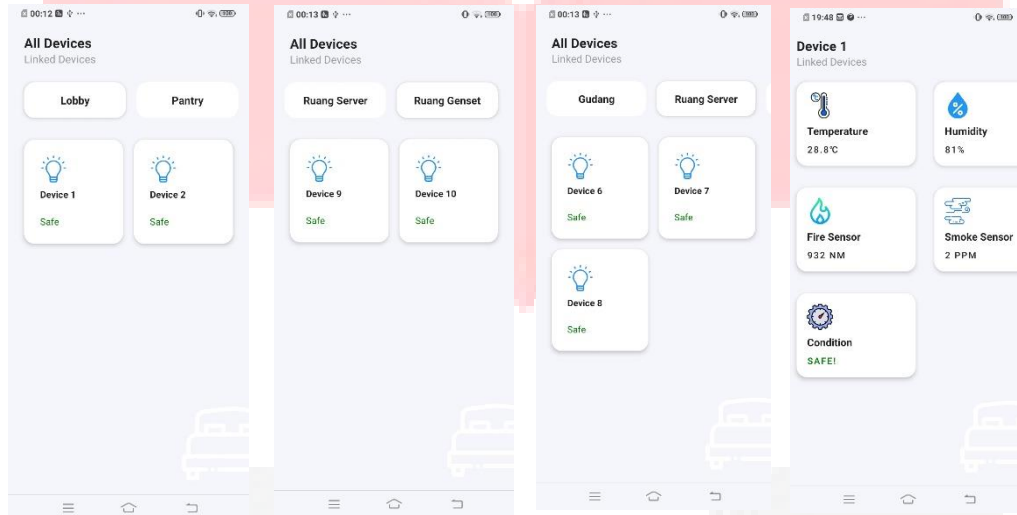
#### 3.2.3 Tampilan Kondisi Aman, Waspada, Bahaya



Gambar 3.4 Tampilan Kondisi Sensor

Ketika kondisi aman aplikasi hanya menampilkan data sensor dari alat secara realtime dan tidak ada notifikasi apapun kepada user atau pemilik gedung. Ketika kondisi waspada, aplikasi akan menampilkan push notification berupa peringatan kepada user atau pemilik gedung. Lalu handphone akan otomatis dalam vibrate mode sampai kondisi menjadi aman kembali. Pada kondisi bahaya, aplikasi akan menampilkan push notification dan tombol untuk memanggil petugas pemadam kebakaran. Dalam kondisi ini handphone user atau pemilik gedung juga dalam keadaan vibrate mode sampai kondisi menjadi aman kembali.

**3.2.4 Tampilan Data Sensor Disetiap Ruang**



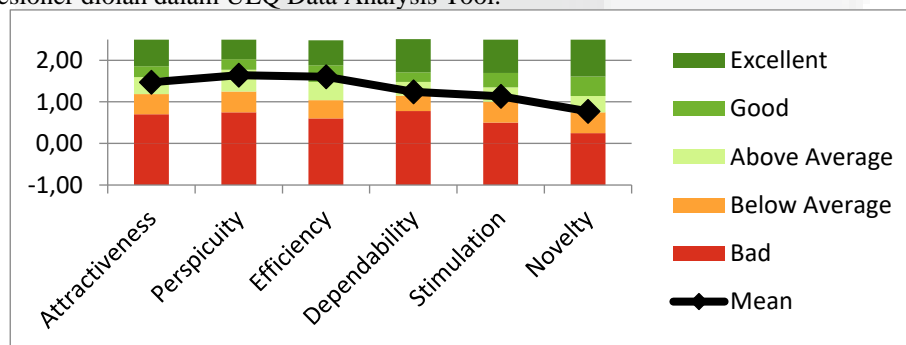
**Gambar 3.5** Tampilan Data Sensor Disetiap Ruang

Pada tampilan ini menampilkan 5 ruangan yang berisikan masing-masing alat. Jumlah sensor setiap ruangan berbeda-beda sesuai dengan luas ruangan dan tingkat keamanan ruangan tersebut. Contohnya seperti Ruang Server memiliki 3 alat karena ruangan tersebut tingkat keamanannya tinggi dan merupakan ruangan yang penting dalam suatu gedung. Data sensor yang ditampilkan disini merupakan dummy data.

**4. Hasil dan Analisis**

**4.1. Pengujian User Experience Questionnaire**

User Experience Questionnaire (UEQ) digunakan untuk membandingkan dan menguji user experience sebuah produk sehingga output yang diharapkan bisa menjadi acuan untuk menentukan bagian mana yang harus diperbaiki atau dikembangkan lebih lanjut. Pada pengujian ini dilakukan dengan mengambil sampel 75 responden menggunakan kuesioner Google Form dan selanjutnya hasil kuesioner diolah dalam UEQ Data Analysis Tool.



**Gambar 4.1** Hasil UEQ

Dari hasil pengujian didapatkan hasil daya tarik sebesar 1.98, kejelasan 1.81, efisiensi 1.83, ketepatan 1.71, stimulasi 1.91 dan kebaruan 1.41. Sehingga dapat disimpulkan bahwa user experience dari aplikasi mobile ini sudah baik.

**Tabel 4.1** Hasil pengujian User Experience dengan UEQ

Scale	Lower Border	Bad	Below Average	Above Average	Good	Excellent	Mean
Daya Tarik	-1,00	0,7	0,49	0,41	0,26	0,64	1,98
Kejelasan	-1,00	0,75	0,5	0,52	0,26	0,47	1,81
Efisiensi	-1,00	0,6	0,44	0,44	0,4	0,6	1,83
Ketepatan	-1,00	0,78	0,37	0,33	0,23	0,8	1,71
Stimulasi	-1,00	0,5	0,5	0,35	0,35	0,8	1,91
Kebaruan	-1,00	0,25	0,5	0,39	0,47	0,89	1,41

Pada hasil pengujian user experience didapatkan data rata-rata terendah pada kebaruan, maka point tersebut bisa menjadi fokus utama untuk pengembangan aplikasi ini selanjutnya.

## 5. Kesimpulan

Dari hasil perancangan sistem, pengujian, dan analisis pada penelitian Tugas Akhir ini, maka penulis mendapatkan beberapa kesimpulan yaitu:

1. Aplikasi mobile berhasil terintegrasi dengan ThingSpeak sehingga mampu menampilkan data secara real time bagi pengguna agar bisa memantau keadaan gedung setiap waktu.
2. Pada pengujian fungsional aplikasi mobile dengan white-box testing dan black-box didapatkan hasil bahwa aplikasi mobile dapat berjalan sesuai dengan algoritma pseudocode dan user interface yang dirancang.
3. Pada pengujian non fungsional didapatkan hasil dari user experience questionnaire dengan poin tertinggi pada skala penilaian daya Tarik sebesar 1.98 dan poin terendah pada skala penilaian kebaruan sebesar 1.41. Dengan rata-rata delay sebesar 7,7 untuk ketiga percobaan pada multi user. Sehingga dapat disimpulkan bahwa user experience dari aplikasi mobile ini sudah cukup baik.

## Daftar Pustaka:

- [1] GUBBI, J., BUYYA, R., MARUSIC, S. & PALANISWAMI, M. 2013. Internet of Things (IoT): A vision, architectural elements, and future directions. *Future generation computer systems*, 29, 1645-1660.
- [2] ATTITALA, S., CHOKSI, V. & POTDAR, M. 2016. Web and Cloud Based Home Automation Systems: An Overview. *International Research Journal of Engineering and Technology*, 3. 1204-1208.
- [3] "Data Detail Kebakaran di Kota Bandung," 2019. [online]. Available: <https://dppk.bandung.go.id>
- [4] JUN, W., DI, Z., MENG, L., FANG, X., HU-LIN, S. & SHU-FENG, Y. Discussion of Society Fire-Fighting Safety Management Internet of Things Technology System. *Intelligent Systems Design and Engineering Applications (ISDEA), 2014 Fifth International Conference on, 2014. IEEE, 422-425.*
- [5] Eltom, R. H., Hamood, E. A., Mohammed, A. A., & Osman, A. A. (2018). *Early Warning Firefighting System Using Internet of Things. 2018 International Conference on Computer, Control, Electrical, and Electronics Engineering (ICCCEEE).*
- [6] Arifiyanto, Farid. 2013. Perancangan prototype web-based online smart home controlled by smartphone. Semarang : Universitas Diponegoro. Hal : 7-8.
- [7] Tam, K., Feizollah, A., Anuar, N. B., Salleh, R., & Cavallaro, L. (2017). The Evolution of Android Malware and Android Analysis Techniques. *ACM Computing Surveys*, Vol. 0, No. 0, Article 00, 1-33.
- [8] Gargenta, M. (2011). *Learning Android*. Sebastopol: O'Reilly

- [9] Khan, M. E., & Khan, F. (2012). *A Comparative Study of White Box, Black Box and Grey Box Testing Techniques*. *International Journal of Advanced Computer Science and Applications*, Vol. 3, No.6, 12-15.
- [10] SAHULEKA, BRIAN., LIM, RESMANA., SANTOSO, PETRUS., Sistem Data Logging Sederhana Berbasis Internet Of Things untuk Pemantauan Suhu Tubuh dan Detak Jantung. Program Studi Teknik Elektro, Universitas Kristen Petra, 32, 1411-870X.
- [11] Arafat, M. K. (2016). SISTEM PENGAMANAN PINTU RUMAH BERBASIS Internet Of Things ( IoT ) Dengan ESP8266. *Jurnal Ilmiah Fakultas Teknik "Technologia,"* 7(4), 262–268.
- [12] Y. D. Yun, C. Lee, and H. S. Lim, "Designing an intelligent UI/UX system based on the cognitive response for smart senior," *Proceeding - 2016 2nd Int. Conf. Sci. Inf. Technol. ICSITech 2016 Inf. Sci. Green Soc. Environ.*, pp. 281–284, 2017
- [13] Khan, M. E., & Khan, F. (2012). *A Comparative Study of White Box, Black Box and Grey Box Testing Techniques*. *International Journal of Advanced Computer Science and Applications*, Vol. 3, No.6, 12-15. [14] R. Syahputra, M. Abdurrohman, and S. P. S. T, "Pendeteksi Kelelahan Untuk Aktivitas Jogging Menggunakan Fuzzy Logic," *Telkom Univ.*, vol. 16, p. 3, 2019.
- [14] R. Wulandari, "Analisis Quality of Service (QoS) Pada Jaringan Internet Studi Kasus : UPT Lokauji Teknik Penambangan Jampang Kulonprogo - LIPI," vol. 2, pp. 162–172, 2016